

Контрольная работа по матрицам

Группа ПИ8

2024-11-16

Умножение матриц

1. Умножение матриц — алгоритм: смотри [Википедию](#)
2. Ввести 2 матрицы и умножить их одну на другую
3. Ввод осуществить из входного файла в строку, аналогично тому, как делали в VPL 5
4. Вывод введённых и рассчитанной матриц осуществить в выходной файл в **матричной форме** — то есть по строкам и столбцам с **минимально понятным комментарием** (название матриц)
5. Считать, что данные вводятся верно (проверок делать не надо)
6. Тип элементов матрицы (int, double..) - на выбор каждого (я рекомендую int как наиболее простой тип для отладки)
7. Все матрицы разместить в **динамической памяти**. Особое внимание обратить на **устранение утечек памяти**
8. **Использовать функции**. Эти функции вызывать из main().
 1. размещения матрицы в динамической памяти
 2. ввода матрицы
 1. можно использовать
 1. индексы $b[i][j]$
 2. или указатели (адресная арифметика) $*(b + i) + j$
 3. вывода матрицы
 1. можно использовать

1. индексы $b[i][j]$
2. или указатели (адресная арифметика) $*(b + i) + j$
4. перемножения матриц
 1. можно использовать
 1. **только указатели** (адресная арифметика) $*(b + i) + j$
5. удаление матриц из динамической памяти
9. Допустимо использовать вспомогательные функции, не упомянутые в п. 8
10. Рекомендуется использовать матрицы небольших размеров (2x3 ...)
11. Проверить, что матрицы можно перемножить, если нет, вывести сообщение и завершить программу с кодом 1.
12. Для проверки по ходу выполнения и результата можно использовать соответствующие онлайн-калькуляторы, например, [этот \(англ\)](#) или [этот \(рус\)](#)
13. Раздельная компиляция не требуется, всё написать в одном .cpp
14. Работы загружать на [dl](#) — если возможно, то с файлами данных
15. Дедлайн: **2024-11-16 18:00**